

Design and Modeling Framework for DexTeR: Dexterous Continuum Tensegrity Manipulator

Cole Woods

Agile Robotics Laboratory,
Department of Mechanical Engineering,
The University of Alabama,
Tuscaloosa, AL 35487
e-mail: cswoods@crimson.ua.edu

Vishesh Vikas¹

Assistant Professor
Mem. ASME
Agile Robotics Laboratory,
Department of Mechanical Engineering,
The University of Alabama,
Tuscaloosa, AL 35487
e-mail: vishesh.vikas@ua.edu

The field of tensegrity faces challenges in design to facilitate efficient fabrication, and modeling due to the antagonistic nature of tension and compression elements. The research presents design methodology, and modeling framework for a human-spine inspired Dexterous continuum Tensegrity manipulator (DexTeR). DexTeR is a continuum manipulator that comprises of an assembly of “vertebra” modules fabricated using two curved links and 12 strings, and actuated using motor-tendon actuators. The fabrication methodology involves the construction of the equivalent graph of the module and finding the Euler path that traverses every edge of the graph exactly once. The vertices and edges of the graph correspond to the holes and strings or links of the mechanism. Unlike traditional rigid manipulators, the design results in centralization of the majority of the weight of the actuators at the base with negligible effect on the manipulator dynamics. For the first time in literature, we fabricate a tensegrity manipulator that is assembled using ten modules to conceptually validate the time and cost efficiency of the approach. A dynamic model of a vertebra module is presented using the Euler–Newton approach with screw theory representation. Each rigid link is represented using a screw, a six-dimensional vector with components of angular rotation, and linear translation. The nonlinearity in the system arises from the discontinuous behavior of the strings and the “closed-chain” nature of the mechanism. The behavior of the strings is piece-wise continuous to model their slack, compliant, or tension states. [DOI: 10.1115/1.4056959]

Keywords: bio-inspired design, robot design, soft robots, tensegrity, continuum manipulator, soft manipulator

1 Introduction

Tensegrity mechanisms synergistically combine *tension* elements (pre-stressed cables) with compression elements (rigid rods) to achieve structural *integrity*. This concept is prevalent from the model of the universe where the compression elements (heavenly bodies) are floating in a sea of tension (gravitational force) to micro-scale biological organisms [1,2]. From an engineering perspective, these mechanisms are packable, portable, internally stable (i.e., do not require gravity for maintaining structural integrity), and possess high strength-to-weight ratios [3,4]. This makes them ideal for applications relating to space, bio-mechanical modeling, and robotic manipulation. These advantages can be viewed as a result of the strategic interaction between the tension-compression elements that preserves the structural integrity of the mechanism. However, this combination of the two antagonistic members poses design and modeling challenges.

Complex tensegrity systems result from assembly of smaller, fundamental primitive units, or tensegrity modules. For example, Ikemoto et al. [5] made a modular tensegrity robot arm that comprised of five modules of four-link tensegrity prisms that used 20 pneumatic cylinders to actuate each rigid member; Ramadoss et al. [6], a modular tensegrity arm that has five modules and used three cables to move the arm; Sabelhaus et al. [7,8], a spine for a quadruped with five vertebrae, three of which were active and change the length of cables using motors; Zappetti et al. [9], an icosahedron modular robot that had three modules. Here, all the modular tensegrity robots use node-to-cable connections between consecutive modules, except for Ref. [9], where the

authors use node-to-node approach to connect the triangular face of consecutive icosahedrons. The use of node-to-node connections when applicable, has potential to tremendously simplify the design.

Despite interest in tensegrity mechanisms since their conception by Buckminster Fuller in the 1960s, their fabrication methodology remains minimally discussed in literature given its arduous nature. It is further complicated with integration of actuators. For example, Refs. [5–7,10] all fabricated tensegrity spine-like structures, however, only Ref. [6] has less actuators than the total number of modules. From the perspective of design, systems with fewer actuators are easier to fabricate. Increasing the number of actuators potentially provides more controllability over the system, e.g., variation in stiffness and shape change. However, this comes at the cost of complexity in design and control (computation power and algorithm). Passively, Zappetti et al. [11] fabricated a tensegrity spine with variable stiffness. Rhodes et al. [12] fabricated a tensegrity robot that could change its shape. Kobayashi et al. [13] made a six-bar modular robot that could change its shape and locomote where each module could move on its own. Despite interest and exploration of tensegrity mechanisms as robot locomotors and manipulators, their fabrication and design methodology is understudied in the literature. This is due to the piece-wise continuous nature of the rigid-tension elements, i.e., sudden change from tension to compression at nodes.

Modeling of the highly non-linear tensegrity systems has been widely investigated to obtain their static equilibrium, commonly termed as form-finding [14]. The use of force density method introduced by Linkwitz and Schek has been instrumental in linearization of the system [15]. Here, the force per unit length in the string allows the expression of the force in a string to be linearly dependent on the position of the start and the end node of the connection. Nevertheless, form-finding is challenging given the discontinuous (tension-compression) and the kinematically closed-chain nature of the problem. In all the methods pursued by researchers, the most widely adopted approach uses a node-based framework to

¹Corresponding author.

Contributed by Mechanisms and Robotics Committee of ASME for publication in the JOURNAL OF MECHANISMS AND ROBOTICS. Manuscript received June 15, 2022; final manuscript received February 10, 2023; published online March 28, 2023. Assoc. Editor: Charles J. Kim.

represent the different nodes of a system and the forces acting upon it [16–20]. The node-based framework has proven to be effective, however, it does have its shortcomings, e.g., the requirement of external forces and torques to be represented in the global coordinate system, and inclusion of additional non-convex, non-linear length constraints in the system dynamics. The former leads to increase in computation and decrease in computation accuracy for including external forces and torques represented in the local/body coordinate system. Similarly, the latter results in increase of computation time for satisfying the non-linear, non-smooth length constraint. Next, the dynamic modeling is performed using different methodologies including principles of virtual work. Abourachid et al. [21] use Lagrangian-approach for deriving the dynamics of a series of X-bar mechanisms that simulates a bird's neck. Ma et al. [18] apply the node-based framework for modeling a tensegrity spine. A lot of researchers use the NASA Tensegrity Robotics Toolkit (NTRT) for modeling, which is based on the aforementioned node-based framework [22]. All these analyses assume the system tensile elements to be pre-stressed strings that are non-compliant, i.e., strings do not have spring-like behavior. Furthermore, despite all these investigations, the dynamic modeling of tensegrity systems remains under-researched due to the difficulties faced to integrate dynamic analysis into the existing modeling frameworks.

Contributions: The research proposes modular design of a Dextrous continuum Tensegrity manipulator (DexTeR) inspired by the human-spine. The DexTeR combines ten vertebrae modules, where each module comprises of two curved rigid elements joined together by 12 strings. These modules are assembled using the vertex-to-vertex methodology. The time-efficient fabrication is achieved by methodically finding the Euler path of the graph corresponding to the module solid, in this case, a J_{84} Johnson solid (snub disphendoid) [23]. This path corresponds to the physical routing path of the string. The shape of the 6DOF DexTeR is controlled using four motor-tendon actuators (MTAs) routed on the outside of the manipulator and controlled using four motors located at the base. The modeling framework of this tensegrity manipulator uses the geometric screw theory (Lie groups) approach for representation. In contrast to the node-based approach, this framework offers advantages of (a) the geometric basis where number of unknowns is proportional to the number of rigid links, (b) being applicable to complex morphologies (including multiple connections at a node), and (c) facilitates ease of representation of forces and torques, with spring-tension model of compliant strings (tensile elements).

The paper is structured as follows: The second section discusses the detailed design and fabrication principles. Next, the design of the DexTeR is discussed. Finally, the modeling framework of the single vertebra module is presented.

2 Design and Fabrication Principles

DexTeR is inspired by the human spinal column and comprises of vertebra as the fundamental building block, referred to as a tensegrity module or primitive, Fig. 1. In fact, the design problems involve synthesis and fabrication of modules, inter-module assembly, and their actuation. Subsequently, the fivefold principles can be summarized as module selection, identification of string paths, pre-stressing and structural stability, inter-modular docking, and actuation. Here, we define three terms for describing a tensegrity structure: a *link* is the rigid and compression element while a *string* or *cable* is the compliant and tension element in the mechanism; and *tendons* are the tension elements that are actuated for affecting shape change. The graph $G(V, E)$ of the mechanism comprises of vertices V and edges E which physically correspond to the connecting holes, and links or strings.

Module selection. A tensegrity module can be visualized as a base polyhedron, e.g., a prism, Platonic, Archimedean, or Johnson solid [24]. Here the edges and vertices of the polyhedron physically correspond to the links or strings, and connection holes. The links may have different morphologies—straight or curved. This selection of

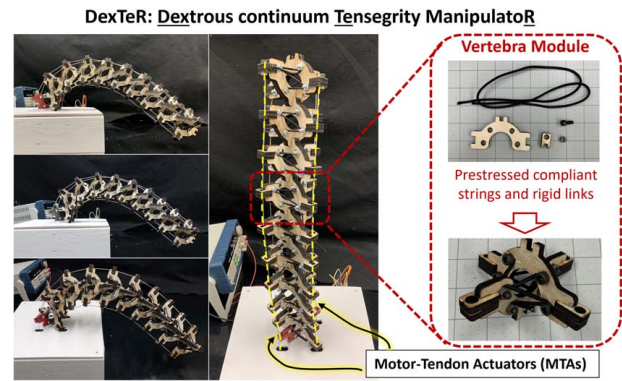


Fig. 1 DexTeR comprises of serially attached vertebrae and controlled using four MTAs—in yellow (dotted line). The fabrication methodology facilitates easy fabrication using pre-stressed strings and two curved rigid links.

link design is critical as, if not chosen appropriately, it may lead to scenarios when the mechanism fabrication is infeasible. Furthermore, link design includes the placement of *connecting holes* where the strings interact with the link. These correspond to the vertices of the aforementioned base polyhedron and graph $G(V, E)$. The shape and placement of these holes play a key role in module stability.

Identification of string paths. Physical testing and experimentation iterations often lead to the desirable mechanism module. Thereby, there is need for a cost- and time-efficient fabrication methodology. For tensegrity mechanisms, this challenge can be distilled to methodology for strategically connecting strings with links. Traditionally, one string per edge is used for connecting two vertices. However, this approach proves to be inefficient as it (a) can be unnecessarily tedious especially as the number of edges per vertex increases, and (b) does not ensure physical stability of the mechanism during the fabrication process, leading to a need for jigs [25]. In contrast, it is much more efficient to identify a set of string paths that use the minimum number of strings. A string path is a sequence of edges between two vertices such that no edge is repeated. In graph theory, this is equivalent to an open or closed trail. This simplifies the process of pre-stressing and tightening of the strings. An example of this could be seen on the vertebra module (discussed later) where one string is used to fabricate the whole mechanism, an Eulerian path or trail in graph theory terminology. This results in no more than two strings at any vertex of the mechanism. In contrast, the one string per edge approach would result in four string ends being tied together and tightened at four different vertices, and two string ends at the other four vertices. However, unlike the vertebra module, not all tensegrity mechanisms will have a string path that can allow the use of only one string. Using graph theory and the well-known problem of the Bridges of Königsberg [26], the number of edges at each vertex must be examined to determine the existence of an Eulerian path. The number of edges at each vertex must be even in order to use one string for the whole mechanism. The reason being, the number of strings entering and exiting the vertex must be the same. For example, a cube has eight vertices and each vertex has three edges, therefore more than one string must be used, in this case, the minimum of four strings. The reason for being, for each of the eight odd vertices, at least one edge enters or exits the node without a paired edge that does the opposite. Since there are two ends to each string, the string is able to exit one vertex and enter another, hence, requiring at least four strings to traverse all the edges. This example illustrates that the minimum number of strings will be half of the number of odd vertices in a mechanism.

Pre-stressing and structural stability. Once the desired set of string paths has been identified, the next step involves effective pre-stressing of the strings. Here, the objective is to facilitate the change of string lengths between vertices in a simple and efficient manner

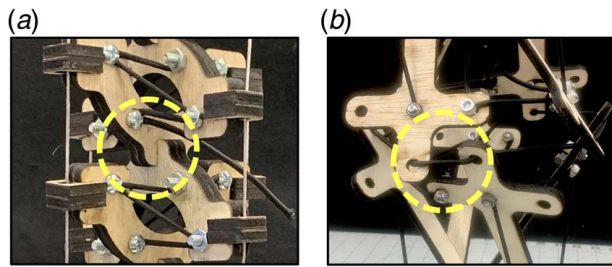


Fig. 2 Inter-modular connections: (a) vertex-to-vertex and (b) string-to-vertex docking

to physically tune the form/shape of the mechanism. The solution methodology incorporates pinch points which are slots off the main hole that pinch the strings and use friction to resist their movement. Pinch points allow for changing the length of one string without compromising the others. Consequently, the string lengths between vertices can be changed until the desired form of the mechanism is achieved. Finally, the strings must be secured to ensure that the pre-stressed strings do not move and the form is not unstable. This can be achieved using multiple approaches, e.g., use of nuts and bolts to tighten the strings, or a pin to resist any motion of the string to leave the pinch point.

Inter-modular docking. The modular design approach affords versatility along with ease and efficiency of fabrication. The concept of *docking* is adopted to simplify connecting one module to another. This can be viewed simplistically when considering rigid modules, however, in the case of tensegrity modules, it needs reconsideration. The concept of docking allows for the internal structure of the module to remain unchanged, however the external part of the module is modified to allow for easier inter-modular connections. The modules can be connected by using two different methods, vertex-to-vertex or string-to-vertex. The former connects a vertex of a link on one module to a vertex of a link on the other module. While the latter is the connection between a vertex on one module and a string on the other. An example of each can be seen in Fig. 2.

Actuation. The strings provide the mechanism with compliance that can be exploited to change the shape. There are multiple ways to actuate the system, in this case, the mechanism is actuated using MTAs. In order to use tendon actuation, a path for the tendon must be designed into the mechanism while also having an origin (motor) and an insertion point (where it connects to the module). From the designer's perspective, the less number of actuators provide ease of fabrication at the cost of control given the high non-linearity of the system.

3 Dexterous Continuum Tensegrity Manipulator

The vertebra, column fabrication, and actuation of the DexTeR are performed using the principles described in the previous section.

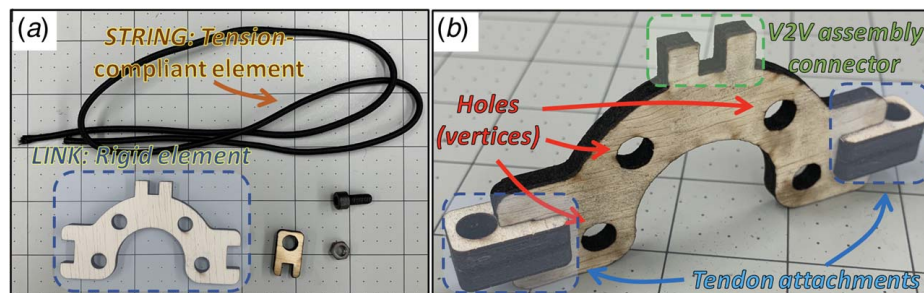


Fig. 3 (a) Elementary fabrication elements include string, link, tendon attachment, and assembly screw and nut and (b) resulting sub-vertebra structure upon integration of the tendon attachments

Module selection and vertebra construction. The tensegrity mechanism based on the snub disphenoid is chosen as the vertebra module of the whole structure. They are fabricated using 1/4" Sande plywood (rigid element), 1/8" elastic nylon cord (tension-compliant element), M5 × 12 mm socket head cap screw, and M5 nut as illustrated in Fig. 3. All the parts fabricated using wood were cut using a laser cutter. The curved link of the vertebra uses two 3" outer diameter semi-circles with an internal diameter of 2", referred to as the sub-vertebra. The sub-vertebra has four holes where the strings are attached to the link. In order to be able to maneuver the vertebra and DexTeR, tendons are connected to each vertebra. This is accomplished by the addition of a tendon attachment. The tendon attachment is a rectangle with the dimension of 7/8" × 1/2" with a hole that allows for passing of the tendon. The tendon attachment is fixed onto the sub-vertebra using a bucktooth design in which a rectangular slot is cut out of one of the narrower sides of the tendon attachment, Fig. 3(b). This bucktooth design allows for a stronger joint by distributing the forces over a larger area. The tendon attachment sustains a large force to affect change in the shape of the manipulator. Consequently, the strength of this joint is essential and two rectangular slots were used for each attachment. The resulting sub-vertebra can be seen in Fig. 3(b).

Identification of string paths. The two sub-vertebrae and their connections can be visualized using a graph corresponding to the snub disphenoid as shown in Fig. 4(a). Here, the green color indicates the string (connected vertices), while red indicates the unconnected vertices, i.e., links. As all the vertices in the graph are even (two or four strings per vertex), there is a single Euler's path that traverses all the green edges (strings). Consequently, the fabrication of this vertebra can be performed using a single string. One particular path of how to run the string is illustrated in Fig. 5. It is worth mentioning to the reader that when fabricating a vertebra prototype, it is good practice to label all the vertices and then run the string along the desired path. This practice will greatly reduce confusion with string routing.

Pre-stressing and structural stability. Upon completion of "stringing" of the mechanism, a bolt can be placed in each hole to secure the string in place. The manipulation of mechanism shape or form can be achieved by adjusting the tensions in the strings. This form-tuning process can be laborious, however it can be simplified by including pinch points in the design as seen in Fig. 6(a). Pinch points are helpful for a couple of reasons. First, the string does not remain in the main hole, so it does not obstruct other strings that are required to be routed through the hole. Additionally, the string can be manipulated in and out of the pinch point in an efficient manner during the form-tuning process. Once the form-tuning process has been completed, the string can be secured by using a nut and bolt. The resulting vertebra upon tightening the strings can be seen in Figs. 6(b)–6(d). We investigate the time taken for a person to fabricate a vertebra by observing this methodology on five different subjects. Each subject was provided with all the materials and were tasked to route the strings and perform form-tuning. On an average, for a subject familiar with the process, this time is between 10 min and 15 min.

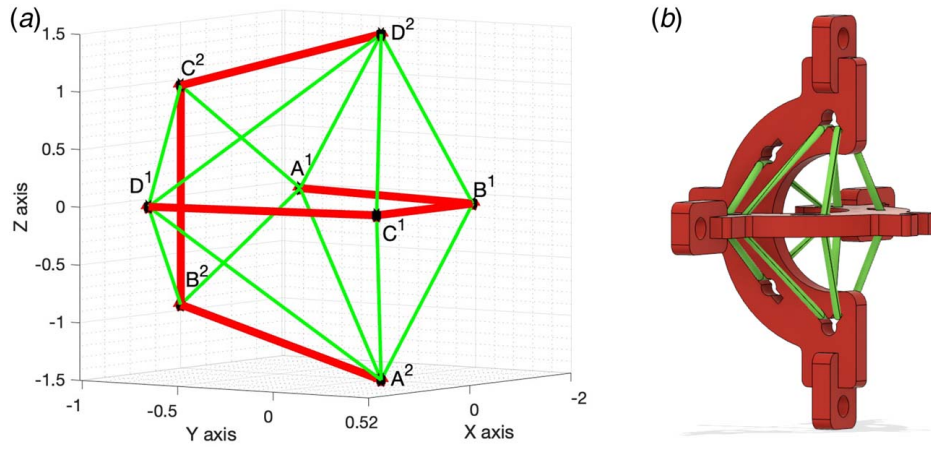


Fig. 4 (a) Base snub disphenoid graph corresponding to the vertebra module where the green (thin lines) indicates the strings (connected edges) and red (thick lines) indicates the two curved links (unconnected edges) having four holes (vertices) A, B, C, D and (b) the CAD model of the vertebra with strings

Inter-modular docking. The manipulator is assembled using the vertex-to-vertex docking approach. The design resembles a bucktooth at the top of the curved link of the sub-vertebra. The reader should notice that the bucktooth is outside the sub-vertebra. This is because in previous versions, the bucktooth was recessed in the sub-vertebra, however this hindered the range of motion of the manipulator column. Each vertebra is attached to the next by using the buckteeth. To strengthen the joint connections of the column, adhesives were used. There are a total of ten connected vertebrae modules in the manipulator column for a total length of 60 cm.

Actuation. Seamless integration of the MTAs into the fabricated continuum manipulator will facilitate shape change.

DexTeR is placed on the base of dimension $12'' \times 12'' \times 4''$. Slots were cut in the base plate that were the same size of the assembly connector of the vertebra. To further support DexTeR, a boat-like support is used to cradle the bottom vertebra. The front and rear sides of the boat-like support cradles the bottom of the sub-vertebra. The base contains four motors housed below the base plate. Along with the slots to house the bottom vertebra, the base plate also contains four slots for the tendons to pass through. The tendons are fixed to the tendon attachments of the top vertebra and run along each of the four sides of DexTeR through the holes in the sub-vertebra bucktooth attachments and end at the motors. The motors are run using an Arduino Mega microcontroller and four

DRV8825 stepper motor drivers. Each MTA is controlled individually to allow for movement of DexTeR. A pose of the DexTeR can be seen in Fig. 7(a).

The wood-based DexTeR of 60 cm has weight of 700 g and is capable of lifting objects. Figures 7(b) and 7(c) show the manipulator lifting a water bottle of approximately same weight as itself. The weight-carrying capability of the robot is dependent on the choice of materials (links and strings), and the actuators (motors).

4 Modeling the Sub-Vertebrae

Modeling of tensegrities has proven to be challenging given the antagonistic nature of the tension-compression elements. Additionally, from the perspective of kinematics, multiple connections between elements result in closed kinematic chains. Finally, the compliant behavior of the strings are piece-wise continuous, incorporating further non-linearity in the system.

As discussed earlier, the most common approach to modeling is the node-based approach [16]. Here, the system unknowns increases with the number of nodes, their application to complex morphologies (e.g., multiple connections at a node) is unknown, the representation of a wrench has to be in the global coordinate system and most importantly, the use of force density to linearly

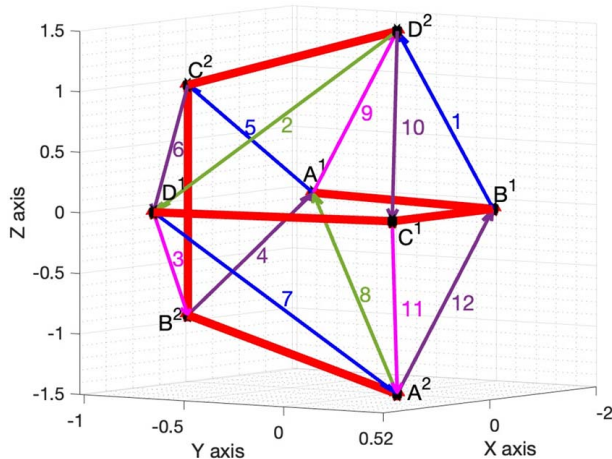


Fig. 5 The Euler path which passes through each edge only once

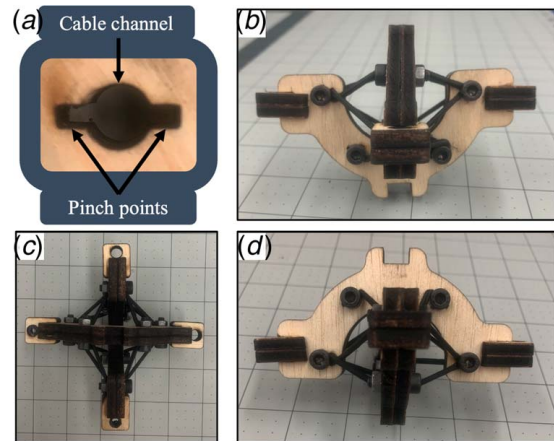


Fig. 6 (a) Hole design that includes the main string channel and the pinch points that aid the form tuning process and (b)–(d) resulting vertebra structure once the string is tightened; front, top, and side views

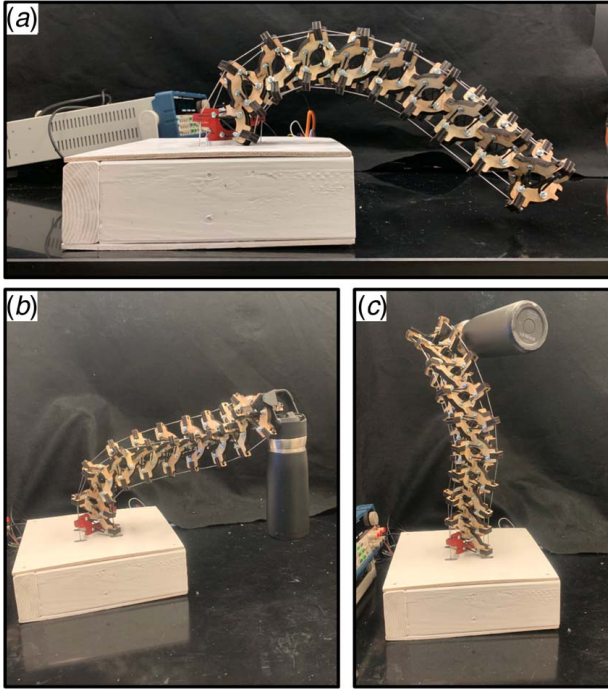


Fig. 7 (a) DexTeR in one of the poses with integrated MTAs. The four motors are housed in the white enclosure below the base plate. The power supply, micro controller, and four motor drivers are placed externally, (b) DexTeR with water bottle sitting on table, and (c) DexTeR with water bottle fully lifted.

model the cables does not incorporate compliance. In contrast, we propose a screw theory approach using Lie groups that circumvents the aforementioned challenges. This is summarized in Table 1. Additionally, the node-based approach models the strings as tensile forces that do not change length, whereas the proposed approach views strings as a spring until the string is stretched to its maximum length where it then becomes pure tension.

4.1 System Description. DexTeR comprises of multiple serially connected vertebrae to construct a virtual open chain. Each vertebra consists of two curved links called sub-vertebrae and 12 strings. All the strings are assumed to have free-lengths $l_{0,j} \forall j = 1, \dots, 12$. Let the relationship between the coordinate systems of the two sub-vertebrae be defined using a screw ξ as shown in Fig. 8. Each sub-vertebra has four connection points or nodes that will be defined as A, B, C , and D . Additionally, each vertex is at some defined distance from the center of mass (CoM) that does not change with time and is consistent for each sub-vertebra.

Each vertebra is serially connected to the neighboring modules via the top and bottom of each vertebra where the top and bottom are defined as the outermost point of each sub-vertebra halfway between node B and C . This connection point will distribute a force between each sub-vertebra. The vertebra will be actuated by

using tendons which run along the outside of each sub-vertebra along nodes A and D . Therefore, there will be a total of 15 forces acting on each sub-vertebra, the 12 string forces, two tendon forces, and single connection force, Fig. 9.

4.2 Framework for Representing Connections. Let the number of vertices and string connections be N_n, N_c respectively. Let the node matrix $P \in \mathbb{R}^{4 \times N_n}$ be the collection of node vectors from the center of mass.

$$P = [p_a, p_b, p_c, p_d, \dots] \in \mathbb{R}^{4 \times N_n} \quad (1)$$

where $p_i \in \mathbb{R}^{4 \times 1}$ is the homogeneous representation of a node point. The string vector matrix $S \in \mathbb{R}^{4 \times N_c}$ is calculated by finding the difference between two nodes on two separate sub-vertebrae. Let the connection matrix $C_i \in \mathbb{R}^{N_n \times N_c}$ for sub-vertebrae i be defined as

$$C_i[j, k] = \begin{cases} 1 & \text{if string } k \text{ contains vertex } j \\ 0 & \text{otherwise} \end{cases} \quad (2)$$

The connection matrices $C_1, C_2 \in \mathbb{R}^{4 \times 12}$ for the presented sub-vertebrae are

$$C_1 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix}^T, \quad C_2 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}^T$$

The transformation matrix $T_{12} \in SE(3)$ that transforms vectors from coordinate system $\{2\}$ to $\{1\}$ is defined using matrix exponential

$$T_{12} = \exp(\hat{\xi}) = \exp(\hat{V}t) \quad (3)$$

where $\xi, V \in \mathbb{R}^{6 \times 1}$ are the screw and twist associated with the two coordinate systems and t is time. The hat operator transforms these vectors to $se(3)$. The reader may refer to Murray et al. [27] with regard to additional details about the notation adopted for this paper.

The string matrix S is the collection of displacement vectors of the strings. As these are free-vectors, the last row of this matrix is always zero. The subscripts 1, 2 associated with S denote the representation of the string displacement vectors in coordinate systems $\{1\}$ or $\{2\}$

$$S_1 = PC_1 - T_{12}PC_2 \quad (4)$$

$$S_2 = PC_2 - T_{12}^{-1}PC_1 = -T_{12}^{-1}S_1 \quad (5)$$

where $S_j \in \mathbb{R}^{4 \times N_c}$. The j th column of the matrix corresponds to the displacement vectors of the j th connection, and the norm is the length of the string.

Table 1 Comparison between traditional approach and one proposed in this research

	Node-based (Traditional)	Lie group/Screw theory (Proposed)
Unknowns (β links)	$3 \times 4\beta = 12\beta$	$6 \times \beta = 6\beta$
Complex morphologies (e.g., multiple connections)	Performance Unknown	Works Very well
Force and torque representation	Global coordinate system (tough to change to local)	Local or global coordinate system (ease of representation)
Non-linear constraints	Length constraints	None
Cable tensile force	Tension	Spring + Tension

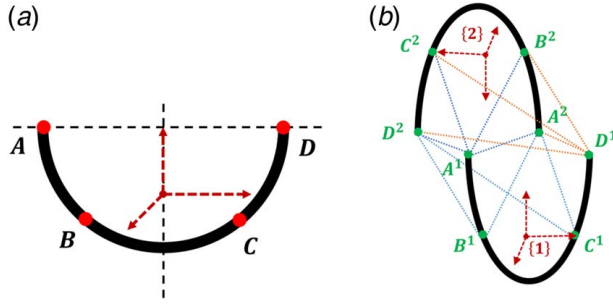


Fig. 8 Each vertebra comprises of two links (semi-circles) and 12 strings (dotted lines). (a) The four element interaction points A, B, C, D on each link are defined with respect to the link coordinate system and (b) two links joined by 12 strings $A^1A^2, A^1B^2, A^1C^2, A^1D^2, D^1A^2, D^1B^2, D^1C^2, D^1D^2, B^1B^2, B^1C^2, C^1B^2, C^1C^2$. The two coordinate systems $\{1\}, \{2\}$ are related by a screw ξ .

$$s_j = \text{col}_j(S), \quad l_j = |s_j| \quad (6)$$

For the rest of the paper, we assume the strings to be firmly fixed between the vertices (holes).

4.3 String Force Model. The compliance in the tensegrity mechanism is imparted by the strings that have spring-tension behavior. The string length to force relationship can be thought about in three regions, Fig. 10: (a) slack—the string length is shorter than the zero free-length (ZFL) l_0 resulting in zero force, (b) spring—the force is generated as per Hooke's law, and (c) tension—force has been exerted on the string to where it has reached its maximum length $l_{\max}$, meaning the force exerted on the string is now a combination of a spring and pure tension force. Consequently, the force in string j now can be defined as

$$f_{s,j} = \begin{cases} 0 & l_j < l_0 \\ k_j(l_j - l_0)/l_j & l_0 < l_j < l_{\max} \\ k_j(l_{\max,j} - l_0 + l_{t,j})/l_j & l_j = l_{\max} \end{cases} \quad (7)$$

where $l_j = |s_j|$, $s_j = \text{col}_j(S)$

where k_j is the linear stiffness of spring j .

The force vector $f_s \in \mathbb{R}^{N_c \times 1}$ and matrix $F_s \in \mathbb{R}^{N_c \times N_c}$ are

$$f_s = [f_{s,1}, f_{s,2}, \dots]^T, \quad F_s = \text{diag}(f_s) \quad (8)$$

Consequently, the wrench due to the string forces, $\mathcal{F}_s$, is

$$\mathcal{F}_s = \begin{bmatrix} \sum_{j=1}^{N_c} f_{s,j} s_j \\ \sum_{j=1}^{N_c} r_j \times (f_{s,j} s_j) \end{bmatrix}, \quad r_j = \text{col}_j(PC) \quad (9)$$

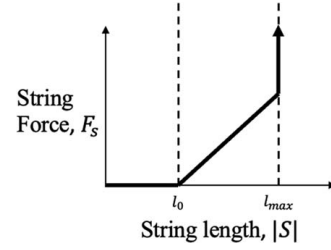


Fig. 10 The string length with respect to the force applied to the string. The plot shows the piecewise nature of the string which introduces nonlinearities into the system.

4.4 Dynamics of a Single Vertebra. We formulate the problem using the Newton–Euler approach that can be later iteratively extended to the whole manipulator with multiple modules. Consider a single rigid sub-vertebra such that the origin is placed at its center of mass, Fig. 8(a), i.e., $\int r \cdot dm = 0$. Assume the sub-vertebrae is moving in space with body twist $\mathcal{V}_b$ that is composed of body angular velocity ω_b and linear velocity v_b of the origin of the body coordinate system expressed in $\{b\}$.

$$\mathcal{V}_b = [\omega_b^T, v_b^T]^T$$

Conservation of linear and angular momentum results in

$$\begin{bmatrix} m_b \\ f_b \end{bmatrix} = \begin{bmatrix} \mathcal{I}_b & 0 \\ 0 & m\mathbf{1} \end{bmatrix} \begin{bmatrix} \dot{\omega}_b \\ \dot{v}_b \end{bmatrix} + \begin{bmatrix} \hat{\omega}_b & 0 \\ 0 & \hat{\omega}_b \end{bmatrix} \begin{bmatrix} \mathcal{I}_b & 0 \\ 0 & m\mathbf{1} \end{bmatrix} \begin{bmatrix} \omega_b \\ v_b \end{bmatrix}$$

$$\mathcal{F}_b = \mathcal{G}_b \dot{\mathcal{V}}_b - (ad_{\mathcal{V}_b})^T \mathcal{G}_b \mathcal{V}_b \quad (10)$$

where m_b, f_b are the moment and force; $m, \mathcal{I}_b$ are the mass and moment of inertia of the rigid link; and $\mathcal{G}_b$ is the spatial inertia matrix, all expressed in body coordinate system.

$$\mathcal{G}_b = \begin{bmatrix} \mathcal{I}_b & 0 \\ 0 & m\mathbf{1} \end{bmatrix}$$

The adjoint of the body twist $ad_{\mathcal{V}_b}$ is defined as

$$ad_{\mathcal{V}_b} = \begin{bmatrix} \hat{\omega}_b & 0 \\ \hat{v}_b & \hat{\omega}_b \end{bmatrix}$$

Here, the hat operator is the skew symmetric operator takes $\mathbb{R}^{3 \times 1} \rightarrow so(3)$. The reader may refer to Refs. [27,28] for details regarding the derivation.

The wrench of the sub-vertebra is the sum of the forces and moments acting on the link. This can be written as

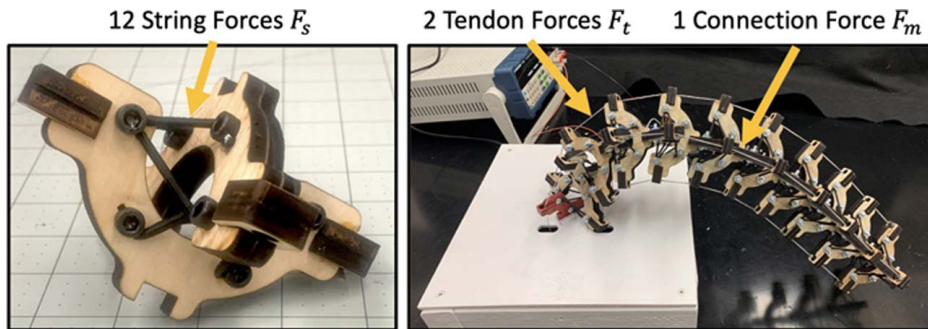


Fig. 9 There are 15 total forces acting on each sub-vertebra. Twelve from the strings, two from the tendons, and one from the connection to the adjoining sub-vertebra.

$$\mathcal{F}_b = \mathcal{F}_s + \mathcal{F}_t + \mathcal{F}_m \quad \text{s.t.} \quad \mathcal{F}_t = \begin{bmatrix} \sum_k F_{t_k} \\ \sum_k \mathbf{r}_{t,k} \times F_{t_k} \end{bmatrix}, \quad \mathcal{F}_m = \begin{bmatrix} F_m \\ \mathbf{r}_m \times F_m \end{bmatrix} \quad (11)$$

where the wrench of the 12 strings $\mathcal{F}_s$ is defined in (9); F_{t_k} , $\mathbf{r}_{t,k}$ is one of the two tendon forces, and the displacement of the tendon from the CoM; and F_m , $\mathbf{r}_m$ is the connection force and the location of the connection from the CoM. The moment arms of the forces for the tendon $\mathbf{r}_{t,k}$, connection point $\mathbf{r}_m$, and strings $\mathbf{r}_{s,j}$ will be known and constant. In case of forward kinematics, the tendon forces can be assumed to be known. As observable, the nonlinearity in the system is introduced through the string wrench term—(a) due to the behavior of the strings (7), and (b) inverse proportionality to the length of the string l_i . It is worth reminding the reader that the frictional interaction between the MTAs and the vertebra can be incorporated in $\mathcal{F}_t$.

These dynamics do not consider damping in the system. The linear viscous damping in the individual string is

$$f_{sd,j} = c_j \frac{d}{dt}(l_j) = c_j \frac{d}{dt}(\sqrt{s_j^T s_j}) = \frac{c_j}{l_j} \left(\mathbf{s}_j^T \frac{ds_j}{dt} \right) \quad (12)$$

where c_j is the string j damping coefficient, which are assembled as a damping vector $\mathbf{c} \in \mathbb{R}^{N_c \times 1}$. This can be incorporated into the system dynamics by modifying the string forces to

$$\tilde{f}_{s,j} = (f_{s,j} + f_{sd,j})\mathbf{s}_j, \quad \text{where } \dot{\mathbf{s}}_j = \text{col}_j(\dot{\mathbf{S}}_2), \quad \dot{\mathbf{S}}_2 = -T_{12}^{-1}PC_1 = T_{12}^{-1}\dot{T}_{12}T_{12}^{-1}PC_1 = \hat{\mathbf{V}}_b T_{12}^{-1}PC_1 \quad (13)$$

To facilitate simulation of these dynamic equations, the state space representation is

$$\begin{bmatrix} \dot{\mathcal{V}}_b \\ \dot{\mathcal{V}}_b \end{bmatrix} = \begin{bmatrix} 0 & I \\ 0 & G_b^{-1}(ad_{\mathcal{V}_b})^T G_b \end{bmatrix} \begin{bmatrix} \xi_b \\ \mathcal{V}_b \end{bmatrix} + \begin{bmatrix} 0 \\ G_b^{-1}\mathcal{F}_b \end{bmatrix} \quad (14)$$

The flow of data for the simulation is visualized in Fig. 11.

4.5 Static Form-Finding. The form-finding problem is formulated as finding ξ such that

$$\mathcal{F}_s + \mathcal{F}_t + \mathcal{F}_{ext} = 0 \quad (15)$$

where the tendon and external wrenches, $\mathcal{F}_t$, $\mathcal{F}_{ext}$, are known. This problem can be re-stated as a root-finding problem of six variables corresponding to ξ .

We examine two numerical examples, with and without external force exerted on a single vertebra. The string stiffness is assumed to be same for all strings as $k = .77$ N/cm with different free-lengths are tabulated in Table 2 where the maximum length is 2 cm longer than each free-length; and identical sub-vertebrae where the vertices from the CoM are $A = [-3.0162, 0, 0]^T$, $B = [-1.5240, -2.6035, 0]^T$, $C = [3.0162, 0, 0]^T$, $D = [1.5240, -2.6035, 0]^T$. The resulting form of the mechanism is visualized

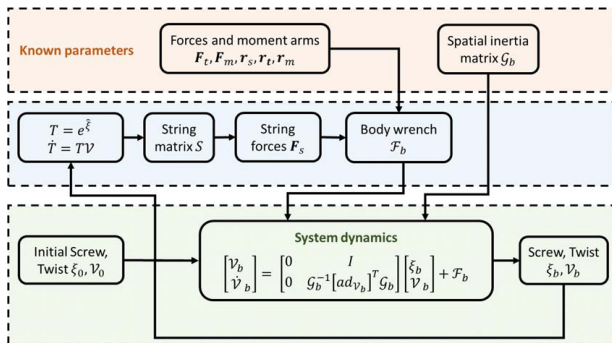


Fig. 11 The block diagram indicating the flow of data for state space dynamics derived using Newton–Euler approach

Table 2 String lengths for the static form-finding simulations when external force is zero and non-zero

String	$F_{ext} = [0, 0, 0]^T \text{N}$		$F_{ext} = -[0, 0.75, 0.5]^T \text{N}$	
	l_0 (cm)	l (cm)	l_0 (cm)	l (cm)
A^1A^2	3.81	4.2955	3.81	4.1893
A^1B^2	3.175	3.9775	3.175	3.4732
A^1C^2	3.175	3.9775	3.175	3.6708
A^1D^2	3.81	4.2955	3.81	4.5105
B^1A^2	3.175	3.9775	3.175	2.8523
B^1D^2	3.175	3.9775	3.175	4.8663
C^1A^2	3.175	3.9775	3.175	3.1701
C^1D^2	3.175	3.9775	3.175	4.5692
D^1A^2	3.81	4.2955	3.81	4.6192
D^1B^2	3.175	3.9775	3.175	4.5084
D^1C^2	3.175	3.9775	3.175	4.1256
D^1D^2	3.81	4.2955	3.81	3.8465

Note: The bold rows indicate the slack state of the string.

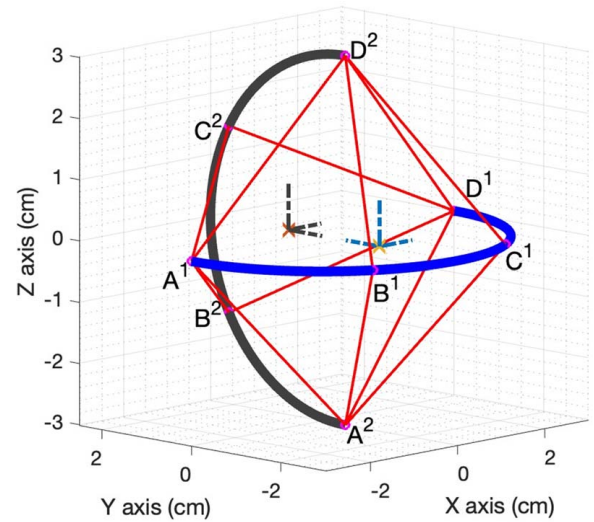


Fig. 12 Simulation of static form-finding where the strings have the slack-spring-tension behavior described in Eq. (7)

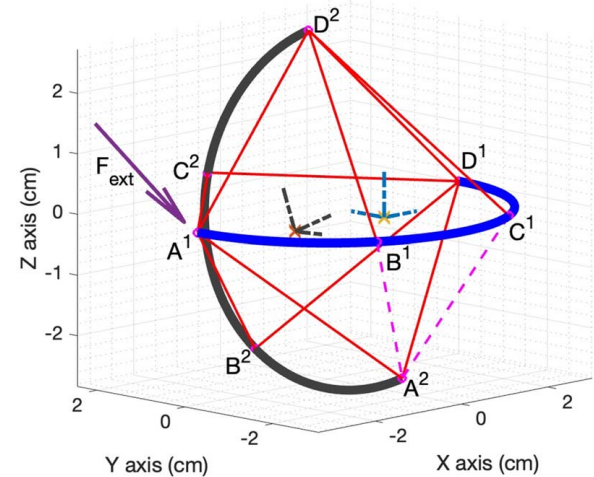


Fig. 13 Form-finding with a non-zero force. Strings A^2C^1 , A^2B^1 (dashed) are slack and have no force as the length between the vertices is less than the ZFL.

in Fig. 12. Here, the screw defining the relationship between the two coordinate systems is $\xi_b = [2.2214, 0, 2.2214, -0.5619, 0, 0.5619]^T$. The coordinates of vertices on curved link 2 can be calculated using the transformation matrix calculated using Eq. (3).

The static form with an external force $F_{ext} [0N, -0.75N, -0.5N]^T$ applied to node A^1 results in the orientation shown in Fig. 13. Here, two of the strings become slack indicated by dashed magenta lines, i.e., the first region in Fig. 10. The body screw $\xi_b = [2.1893, 0.6886, 1.8180, -0.5224, -0.0322, 0.6336]^T$. Such scenarios are observed during experiments when multiple strings become slack upon application of force on the mechanism.

5 Conclusion and Future Works

The paper introduces a human-spine inspired DexTeR that is controlled using four MTAs. The modular design approach facilitates assembly of ten “vertebra” modules comprising of two links and 12 strings. The fabrication approach involves finding the Euler path on the equivalent graph of the module—a Johnsons J_{84} solid. Here, the edges represent the string connections and the Euler path traverses each edge once. Subsequently, the fabrication of a module is accomplished by routing a single string. Design modifications to the mechanism are incorporated to ensure the structural stability and tuning ability of the mechanism. The resulting semi-rigid, continuum manipulator is capable of conforming to desirable shapes to accomplish tasks in its workspace. Unlike traditional rigid manipulators where actuators are located along the arm, location of the MTAs at the base results in centralization of the majority of the weight of the actuators with negligible effect on the manipulator dynamics. A single module of DexTeR is modeled using the Newton–Euler approach with screw theory representation. The proposed framework models the non-linear behavior of the compliant strings, and allows for modeling of complex morphologies. The dynamics of a single vertebra are modeled and discussed. This also includes damping due to strings. The numerical examples for form-finding include scenarios with and without external force acting on the mechanism. The simulation does indicate slack-behavior of a couple of springs upon application of external force as observed during physical interaction with the module.

The future work involves extending the modeling methodology to multiple modules as an iterative approach to model the complete manipulator. The simulation of DexTeR and its physical open-loop control for validation are the logical steps. Additionally, comparison of the proposed framework with the traditional node-based approach needs to be examined.

Acknowledgment

This work is supported by the National Science Foundation under Grant No. 1832993.

Conflict of Interest

There are no conflicts of interest.

Data Availability Statement

The datasets generated and supporting the findings of this article are obtainable from the corresponding author upon reasonable request.

Nomenclature

- P = node matrix, $\in \mathbb{R}^{4 \times N_n}$
- S = string matrix, $\in \mathbb{R}^{4 \times N_c}$
- $f_{s,j}$ = magnitude of the force in the j th string
- k_j = stiffness of the j th string
- l_j = length of the j th string

- $l_{0,j}$ = free-length of the j th string
- s_j = string vector of j th string
- $r_{s,j}$ = displacement vector of the string j from the CoM
- C_i = connection matrix for link i , $\in \mathbb{R}^{N_n \times N_c}$
- N_n = number of vertices on a rigid link
- N_c = number of string connections between vertebrae links
- T_{ab} = transformation matrix between coordinate systems $\{a\}$ and $\{b\}$, $\in \mathbb{R}^{4 \times 4}$
- $se(3)$ = Lie algebra of the Lie group $SE(3)$, $\in \mathbb{R}^{4 \times 4}$
- $so(3)$ = Lie algebra of the special orthogonal group of dimension 3, $\in \mathbb{R}^{3 \times 3}$
- $SE(3)$ = special Euclidean group of dimension 3, $\in \mathbb{R}^{4 \times 4}$
- ξ = screw associated with two coordinate systems, $\in \mathbb{R}^{6 \times 1}$

References

- [1] Motro, R., 2003, *Tensegrity: Structural Systems for the Future*, Butterworth-Heinemann, Oxford, UK.
- [2] Ingber, D. E., 1993, “Cellular Tensegrity: Defining New Rules of Biological Design that Govern the Cytoskeleton,” *J. Cell Sci.*, **104**(3), pp. 613–627.
- [3] Connelly, R., 2002, “Tensegrity Structures: Why Are They Stable?” *Rigidity Theory and Applications. Fundamental Materials Research*, M. F. Thorpe, and P. M. Duxbury eds., Springer, Boston, MA, pp. 47–54.
- [4] Skelton, R. E., Montuori, R., and Pecoraro, V., 2016, “Globally Stable Minimal Mass Compressive Tensegrity Structures,” *Compos. Struct.*, **141**, pp. 346–354.
- [5] Ikemoto, S., Tsukamoto, K., and Yoshimitsu, Y., 2021, “Development of a Modular Tensegrity Robot Arm Capable of Continuous Bending,” *Front. Rob. AI*, **8**, p. 11.
- [6] Ramadoss, V., Sagar, K., Ikbal, M. S., Calles, J. H. L., Siddaraboina, R., and Zoppi, M., 2022, “Hedra: A Bio-Inspired Modular Tensegrity Robot With Polyhedral Parallel Modules,” *IEEE 5th International Conference on Soft Robotics (RoboSoft)*, Edinburgh, UK, Apr. 4–8, pp. 559–564.
- [7] Sabelhaus, A. P., van Vuuren, L. J., Joshi, A., Zhu, E., Garnier, H. J., Sover, K. A., Navarro, J., Agogino, A. K., and Agogino, A. M., 2018, *arXiv:1804.06527 (pre-print)*.
- [8] Sabelhaus, A. P., Ji, H., Hylton, P., Madaan, Y., Yang, C., Agogino, A. M., Friesen, J., and SunSpiral, V., 2015, “Mechanism Design and Simulation of the Ultra Spine: A Tensegrity Robot,” *International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, Vol. 57120, American Society of Mechanical Engineers, p. V05AT08A059.
- [9] Zappetti, D., Mintchev, S., Shintake, J., and Floreano, D., 2017, “Bio-Inspired Tensegrity Soft Modular Robots,” *6th International Conference, Living Machines*, Stanford, CA, July 26–28, Springer, pp. 497–508.
- [10] Fasquelle, B., Furet, M., Khanna, P., Chablat, D., Chevallereau, C., and Wenger, P., 2020, “A Bio-Inspired 3-DOF Light-Weight Manipulator With Tensegrity X-joints,” *IEEE International Conference on Robotics and Automation (ICRA)*, Paris, France, May 31–Aug. 31, pp. 5054–5060.
- [11] Zappetti, D., Arandes, R., Ajanic, E., and Floreano, D., 2020, “Variable-Stiffness Tensegrity Spine,” *Smart Mater. Struct.*, **29**(7), p. 075013.
- [12] Rhodes, T., Gotberg, C., and Vikas, V., 2019, “Compact Shape Morphing Tensegrity Robots Capable of Locomotion,” *Front. Rob. AI*, **6**.
- [13] Kobayashi, R., Nabae, H., Endo, G., and Suzumori, K., 2022, “Soft Tensegrity Robot Driven by Thin Artificial Muscles for the Exploration of Unknown Spatial Configurations,” *IEEE Rob. Autom. Lett.*, **7**(2), pp. 5349–5356.
- [14] Tibert, A., and Pellegrino, S., 2003, “Review of Form-Finding Methods for Tensegrity Structures,” *Int. J. Space Struct.*, **18**(4), pp. 209–223.
- [15] Linkwitz, K., and Schek, H.-J., 1971, “Einige bemerkungen zur berechnung von vorgespannten seilnetzkonstruktionen,” *Ingenieur-archiv*, **40**(3), pp. 145–158.
- [16] Skelton, R. E., and de Oliveira, M. C., 2009, “Analysis of Tensegrity Dynamics,” *Tensegrity Systems*, Springer US, Boston, MA.
- [17] Goyal, R., and Skelton, R. E., 2019, “Tensegrity System Dynamics With Rigid Bars and Massive Strings,” *Multibody Syst. Dynam.*, **46**(3), pp. 203–228.
- [18] Ma, S., Chen, M., Peng, Z., Yuan, X., and Skelton, R. E., 2022, “The Equilibrium and Form-Finding of General Tensegrity Systems With Rigid Bodies,” *Eng. Struct.*, **266**.
- [19] Murakami, H., 2001, “Static and Dynamic Analyses of Tensegrity Structures. Part 1. Nonlinear Equations of Motion,” *Int. J. Solids. Struct.*, **38**(20), pp. 3599–3613.
- [20] Sultan, C., “Modeling, Design, and Control of Tensegrity Structures With Applications,” Ph.D. dissertation, Purdue University, West Lafayette, IN.
- [21] Abourachid, A., Böhmer, C., Wenger, P., Chablat, D., Chevallereau, C., Fasquelle, B., and Furet, M., 2019, “Modelling, Design and Control of a Bird Neck Using Tensegrity Mechanisms.”
- [22] “Tensegrity Robotics – NTRT – NASA Tensegrity Robotics Toolkit”
- [23] Weisstein, E. W., Snub Disphenoid. Wolfram Research, Inc.
- [24] Gotberg, C., and Vikas, V., 2022, “Synthesis of Tensegrity Primitives Using Polyhedron and Lexicographic Ordering,” *ASME J. Mech. Rob.*
- [25] Hudson, T., 2022, <https://trmm.net/Tensegrity/>
- [26] Euler, L., 1953, “Leonhard Euler and the Koenigsberg Bridges,” *Sci. Am.*, **189**(1), pp. 66–72.
- [27] Murray, R. M., Li, Z., and Sastry, S. S., 2017, *A Mathematical Introduction to Robotic Manipulation*, CRC Press, Boca Raton, FL.
- [28] Lynch, K. M., and Park, F. C., 2017, *Modern Robotics: Mechanics, Planning, and Control*, 1st ed., Cambridge University Press, Cambridge.